REMARKS

Entry of this amendment under 37 CFR §1.116 is respectfully requested.

Independent claims 1, 12 and 19 are amended to simplify issues on appeal. In particular, the recital of "recording the voice message using an executable browser plug-in resource configured for" is not necessary and redundant in view of the prior recital of "recording by an executable browser plug-in resource". Hence, entry of this amendment is respectfully requested to minimize issues on appeal.

Claims 26-29 have been amended in view of the objection to use of quotations and the recital of MIME types having a value to them. In particular, claims 26 and 29 as amended specify that the MIME type has a MIME type *extension* of one of .711, .729, and .GSM, as specified on page 11, lines 3-4 of the specification.

The objection to claim 10 as a substantial duplicate of claim 8 is respectfully traversed. Claim 8 specifies a recorder configured for recording a voice message, whereas claim 10 further limits claim 8 by specifying that the recorder includes an executable plug-in resource having executable code including instructions for performing the encoding. Hence, claim 8 reads on any implementation of a recorder performing the claimed operations, whereas claim 10 requires the recorder to be implemented to include an executable plug-in resource configured for performing the claimed encoding. Hence, since claim 10 further limits claim 8 under 35 USC 112, fourth paragraph, claim 10 is proper.

The §103 Rejection in view of Picard and Luzeski

Claims 1-2, 5-8, 10, 12-13, 16-20, and 23-29 stand rejected under 35 USC 103 in view of US Patent No. 6,233,318 to Picard in view of US Patent No. 6,301,245 to Luzeski. This rejection is respectfully traversed.

Each of the independent claims specifies that ***a user computer*** sends a voice message by recording a voice message spoken by a calling party based on encoding the voice message according to any one of G.711, G.729, and GSM encoding protocols. Further, the voice message is stored by the user computer within a data file having a selectable MIME type that is ***recognizable by the voice***

messaging system as a voice message, and wherein ***the MIME type identifies the one encoding protocol*** (i.e., one of the G.711, G.729, and GSM encoding protocols). The data file is output by the user computer using a prescribed messaging protocol for transfer to a destination voice mailbox accessible by the voice messaging system.

Hence, each of the independent claims specifies that the MIME type is recognizable by the voice messaging system as a voice message, and that the MIME type will identify the one encoding protocol used to encode the voice message, namely G.711, G.729, or GSM encoding protocol.

Hence, a voice messaging system is able to recognize the data file as containing a voice message ***based solely on the MIME type*** that identifies the one encoding protocol; moreover, the encoding of the voice message according to any one of G.711, G.729, and GSM encoding protocols (as opposed to conventional PC based encoding protocols) eliminates the necessity for converting (i.e. transcoding) the encoded voice message between a conventional 64 kbps PC format and the 8kbps voice messaging format used in voice messaging systems. These and other features and advantages are neither disclosed or suggested in the applied prior art.

Picard

Applicant strenuously traverses the tortured interpretation of Picard: there is no disclosure whatsoever or any suggestion that Picard provides any MIME type that can be recognizable by the voice messaging system as a voice message. Rather, Picard teaches that the criteria for deciding whether a given MIME type should be recognized by a voice messaging system as a voice message is not evaluation of the MIME type, but rather the recipient address.

In particular, the integrated messaging system of Picard provides a voice interface for conventional telephony-based access with a telephone 102 via the public switched telephone network 104 to an integrated messaging system 106 (col. 6, lines 10-15, col. 8, lines 32-58). Hence, a user accessing the integrated messaging system 106 using a telephone 102 via the PSTN 104 may leave a voice message, causing the integrated messaging system 106 to store the voice message in a "native voice" format (see, e.g., col. 13, lines 41-42).

Picard also describes accessing the integrated messaging system using a PC. Unlike the

telephone-based access where a voice message recorded over the voice interface is stored according to a native voice format, however, any message received by the integrated messaging system via its SMTP or HTTP user interface is stored in the received MIME format (col. 11, lines 28-37 and col. 13, lines 33-58).

Moreover, Picard explicitly specifies that a message is recorded in its native voice or facsimile only if it is received using the DTMF interface (col. 13, lines 41-43).

Further, if a message is received via the SMTP or HTTP user interface, the message is stored as a MIME audio or facsimile type, ***without any recognition as to whether the message contains a voice message*** (as opposed to any other audio information). In particular, Column 13, lines 41-57 clearly teaches that a message is recognizable by the voice messaging system as a voice message only if received using the DTMF interface, and that any other message received via the SMTP or HTTP user interface **cannot** be considered as a voice message unless the recipient address is a phone number:

> a. If the message is native voice or facsimile **(i.e., using the DTMF interface)**, and the recipient address is a phone number, the message is handled in the conventional VMS manner for a voice or facsimile message.
>
> b. If the message is native voice or facsimile, and the recipient address is not a phone number, the message is sent to the EMS 66, with the data converted to a MIME audio or image/tiff type.
>
> c. If the message is a MIME audio or facsimile type (i.e., SMTP or HTTP user interface), ***and the recipient address is a phone number***, the data is converted to native format and handled as conventionally by the VMS (note: addressing to a phone number could be prohibited to avoid this conversion, initially).
>
> d. If the message is any MIME type, ***and the recipient address is not a phone number***, the message is sent unchanged to the EMS 66.

As stated in the last step "d.", "if the message is any MIME type, and the recipient address is not a telephone number, the message is sent unchanged to the EMS 66."

Hence, the assertion in the Official Action that the MIME type is recognizable by the voice messaging system as a voice message is incorrect and without foundation. Rather, Picard does not

use the MIME type to recognize a messages containing a voice message, but rather relies on ***the messaging source*** (e.g., DTMF interface) or ***the destination*** (recipient address being a phone number) to determine whether a message contains a voice message.

Applicant further traverses the Examiner's assertion of Official Notice that the added limitations of "encoding the voice message according to any one of G.711, G.729, and GSM encoding protocols... where the MIME type identifies the one encoding protocol" is "old and well-known." The Examiner refers to MIME encoding in general, without any support for the *specific* assertion of ***encoding*** the voice message **by the user computer** according to the specified encoding protocols. The citation of USP 6,549,612 to Gifford is misplaced, because column 15, lines 10-14 simply describe the supported formats that may be used **by a messaging server** sending a message to a destination subscriber.

There is no disclosure or suggestion, however, of a ***user computer*** encoding the voice message according to the specified protocols.

No Motivation to Modify Picard to Include Teachings of Luzeski

As admitted in the Official Action, Picard fails to disclose or suggest the recording step which includes encoding the voice message of according to one of the G.711, G.729, and GSM encoding protocols. The Official Action fails to provide any motivation to modify the primary reference to include the teachings of Luzeski. "The mere fact that the prior art may be modified in the manner suggested by the Examiner does not make the modification obvious unless the prior art suggested the desirability of the modification." In re Fritch, 23 USPQ2d 1780, 1783-84 (Fed. Cir. 1992). In re Mills, 16 USPQ2d 1430 (Fed. Cir. 1990).

In particular, the supposed motivation in paragraph 3 on page 4 of the Office Action is both improper, without foundation, and nonsensical. The use of the applicant's specification is per se improper: "It is impermissible to use the claimed invention as an instruction manual or 'template' to piece together the teachings of the prior art so that the claimed invention is rendered obvious." In re Fritch, 23 USPQ2d 1780, 1784 (Fed. Cir. 1992).

Further, the reference to the G.711, G.729, and GSM standards as set by the ITU community

is a non sequitur, especially since it disregards the purpose for the standards being set by the ITU committee, namely recording messages within a voice messaging system.

In fact, Picard provides numerous teachings that only conventional PC based audio systems should be used (see, for example, col. 15, lines 14-20 which specifies use of a conventional soundcard and conventional hardware such as Sound Blaster from creative, Inc., and media player software from Microsoft, and col. 18, line 65 to col. 19, line 25, which suggests that the recording can be performed by a conventional plug-in or at the server.)

Hence, contrary to the Examiner's suggestions, there is no evidence of any motivation for once skilled the art to modify the teachings of Picard to have included encoding voice messages at a rate of 8 kHz, as asserted.

Moreover, one having ordinary skill in the art would not have been motivated to modify Picard in order to include the teachings of Luzeski, as asserted. In particular, the Examiner relies on a piecemeal application of Luzeski, relying solely upon ***eight lines of the appendix***, while completely disregarding the entire teachings of Luzeski. Luzeski must be considered in its entirety, i.e., as a whole, including portions that would lead away from the claimed invention (see MPEP 2141.02 at page 2100-95 (Rev. 1, Feb. 2000) (citing W.L. Gore & Associates, Inc. v. Garlock, Inc., 22 USPQ 303 (Fed. Cir. 1983), cert. denied, 469 U.S. 851 (1984))).

Luzeski

Luzeski does not teach or suggest storing a voice message having a MIME type recognizable as a voice message and identifying the encoding protocol ***in the user computer***, as claimed, but rather relies on intermediate nodes **between** the user computer and the message store to create the necessary data structures to identify the recorded message as a voice message.

Luzeski stresses that unified messaging services are provided by a content manager 12-2 (see Figure 1) that is designed to receive information from content providers, format information into multimedia containers, and distribute the multimedia containers to the universal messaging system via CMC Applicaton Programming Interface calls (col. 5, lines 18-30, and col. 12, line 45 to col. 13, line 20).

The content manager 12-2 in the web-based server platform 12 is responsible for the generation and maintenance of the content messages flowing in and out of the universal messaging system based on issuing CMC calls, VNMS support library calls, and requests to the messaging platform file system (col. 13, lines 48-53.). The content manager 12-2 in the web-based server platform 12 relies on the session manager 10-5 in the messaging platform 10 to provide a persistent session with the CMC API 10-4, which requires establishment of a persistent session on behalf of the content manager 12-2 (see col. 6, line 42 to col. 7, line 15 and col. 7, lines 53-59).

Hence, the content manager 12-2 issues procedure calls to the CMC API 10-4 (via the session manager 10-5) (e.g., col. 9, lines 53-60) in order for **the CMC API 10-4 to create a CMC message structure**, described with respect to Fig. 2 at col. 14, line 5 to col. 18, line 21.

Moreover, Luzeski teaches that audio portions should not be directly attached to message structures, rather they should be stored in the voicemail message store 10-9 (within the NAP Telephony Tools), and that the message attachment should instead include in the content message a pointer to the audio stored in the message store 10-9 (see col. 16, lines 36-52, and col. 17, lines 28-32).

Hence, the Examiner's reference to the appendix actually would be interpreted by one skilled in the art as content object type **defined by the CMC API 10-4 generating a data structure** that identifies the location of the voice message in the voice message store 10-9: this interpretation is consistent with additional descriptions in the reference of a user retrieving information about a voice message stored in the voice storage 10-9 (col. 18, lines 29-32) based on first opening an e-mail notification of the voice message via a path "A" of Fig. 1 (see Fig. 4C and col. 19, line 65 to col. 20, line 20), followed by actual retrieval of the voice message via the secondary path "B" via the web server 14 (see Fig. 1 and col. 12, lines 1-26, and Fig. 4D and col. 20, lines 22-45).

Consequently, Luzeski requires a complex data structure to be generated by the CMC API 10-4 *within the messaging platform 10* based on a session-based interaction between the CMC API 10-4 and the content manager 12-2 within the web platform 12, which receives the encoded voice message from the user PC 20 via the Internet 16 (see Fig. 4G and col. 21, lines 20-43).

Moreover, Luzeski explicitly specifies under the heading "Message Types" starting at col.

14, line 33 that "a new series of message types specifically targeted for the delivery of multimedia content" will "be **ignored by normal e-mail clients** but will be visible to the CMC layer 10-4 clients *specifically coded to recognize the message type*", namely a "custom client" such as the UVMS Custom Client 10-7 that includes a CMC client architecture capable of referencing CMC APIs and that has additional logic to handle the extended range of type fields required for multimedia containers (see, e.g., col. 13, lines 1-3 and col. 17, line 40 to col. 18, line 21).

Hence, Luzeski requires the construction of data structures according to a CMC API using "custom clients", illustrated as residing **within the messaging platform 10**, in order to use the object content types identified in the Appendix to identify the message as a voice message.

It is not until after the data structures have been completed by the CMC API that the voice message can be stored via a streaming connection with the web server 14 (see detailed comments *infra* regarding the §103 rejection in view of Luzeski, citing Fig. 4G and col. 21, lines 20-43 and quoting col. 12, lines 1-26).

Each of the independent claims, however, specify that ***the user computer*** stores the voice message within a data file having a selectable MIME type recognizable by the voice messaging system as a voice message, the MIME type identifying the one encoding protocol.

Consequently, the hypothetical combination neither discloses nor suggests ***a user computer*** that encodes the voice message according to any one of G.711, G.729, and GSM encoding protocols, and ***stores*** the voice message ***within a data file*** having a selectable MIME type recognizable by the voice messaging system as a voice message ***and identifying the one encoding protocol.***

For these and other reasons, this rejection in view of Picard and Luzeski should be withdrawn.

The §103 Rejection in view of Luzeski

The rejection of claims 1-2, 5-8, 12-13, 16-20, and 23-29 in view of Luzeski is respectfully traversed. The comments above describing Luzeski are incorporated in their entirety herein by reference.

The reliance of column 20, lines 32-38 as teaching the storage of a voice message within a data file having a selectable MIME type recognizable by the voice messaging system as a voice message is misplaced, since the cited portion refers to the retrieval of a voice message from the voice messaging system, and ***not*** the storage of a voice message by a ***user computer***. Hence, the Examiner is arguing the exact opposite of what is being taught, namely that the teaching of *retrieving* voice messages from the messaging system (as disclosed in Luzeski) should read on the claimed ***sending*** of a voice message ***by a user computer*** to a messaging system.

As such, the Official Action fails to provide any disclosure or suggestion whatsoever that the claimed storing of a voice message within a data file having a selectable MIME type ***that is recognizable by the voice messaging system as a voice message***, and wherein the MIME type ***identifies the one encoding protocol***, is performed ***by the user computer sending the voice message.***

As described above with respect to the §103 rejection in view Picard and Luzeski, Luzeski requires the CMC API 10-4 in the messaging platform 10 to perform the necessary session-based generation of the requisite data structures to identify in an e-mail that the message to be transmitted is a voice message (see Fig. 4G and col. 21, lines 20-43). This data structure, however, does not store the voice message, but only **describes attributes about the storage of the voice message**.

Storage of the actual voice message requires a distinct operation, namely a streaming connection between the user computer 20 and the web server 14 of Fig. 1, which issues a function call to session manager 10-5 as a CGI library to access the VMMM 10-8 interface library 10-8:

> Streaming of voice and fax data both into and out of the VMMM Voice/Fax Store 10-9 is accomplished with the Web Server 14 which calls Session Manager 10-5 as a CGI library using the CGI_Session_Manager entry point. Two procedures, one for input and one for output, handle all streaming requirements.
>
> The Session Manager's CGI interface ... must be able to accept a data stream recorded by the plugin and representing a voice or fax message and process it appropriately.
>
> ...
> After recording a voice message, the plugin streams data to the Session Manager which invokes a procedure called Accept Voice Fax. The Session Manager then calls the VMMM

10-8 interface library to store the data.

(Col. 12, lines 1-26).

There is no disclosure whatsoever that the voice message is stored ***by the user computer*** in a data file having a selectable MIME type recognizable by the voice messaging system.

An evaluation of obviousness must be undertaken from the perspective of one of ordinary skill in the art addressing the same problems addressed by the applicant in arriving at the claimed invention. Bausch & Lomb, Inc. v. Barnes-Hind/Hydrocurve, 23 USPQ 416, 420 (Fed. Cir. 1986), cert. denied, 484 US 823 (1987). Thus, the claimed structures and methods cannot be divorced from the problems addressed by the inventor and the benefits resulting from the claimed invention. In re Newell, 13 USPQ2d 1248, 1250 (Fed. Cir. 1989).

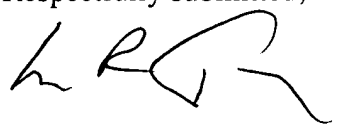
None of the applied references in any of the foregoing rejections, singly or in combination, address the advantage of a single user computer being able to generate a message that enables a voice messaging system to recognize the message as a voice message ***based solely on the MIME type that identifies the encoding protocol***, as claimed. Rather, the applied references rely on additional parameters (messaging source or messaging destination in Picard, CMC API data structures) that are executed **within the messaging server** to determine whether a received message includes a voice message.

For these and other reasons, these rejections should be withdrawn.

In view of the above, it is believed this application is and condition for allowance, and such a Notice is respectfully solicited.

To the extent necessary, Applicant petitions for an extension of time under 37 C.F.R. 1.136. Please charge any shortage in fees due in connection with the filing of this paper, including any missing or insufficient fees under 37 C.F.R. 1.17(a) or 1.17(e), to Deposit Account No. 50-1130, under Order No. 95-454, and please credit any excess fees to such deposit account.

Respectfully submitted,



Leon R. Turkevich
Registration No. 34,035

Customer No. 23164
(202) 261-1059
Date: September 1, 2005